

In particular, the server 120 is a data processing mechanism for facilitating data exchanges between client data processing systems and a plurality of web page servers 132, 134, and 136. See Tsimelzon at col. 4, lines 8-24. The web page servers 132, 134, and 136 store one or more web pages and the server 120 "stores selection information 122 describing the shortpages¹." See Tsimelzon at col. 4, lines 24-41. When a user requests to view a shortpage, the server 120 identifies the web page needed from web servers 132, 134, and 136 based on the selection information, retrieves the web page, and shortens the retrieved web page based on the selection information prior to providing the retrieved content to the user. See Tsimelzon at col. 4, lines 42-50. In one implementation, the server 120 also stores notification information corresponding to user-selected blocks within a particular web page. See Tsimelzon at col. 11, lines 31-63. In processing a notification, the server 120 waits a specified time interval, retrieves a web page for which a notification condition exists, locates a block corresponding to the notification condition using the selection information stored on the server, determines whether the content of the located block in the retrieved web page meets the notification condition, and notifies the user if the condition is met. See Tsimelzon at col. 12, lines 46-67.² As such, server 120 is instructed to generate and send a notification to a user in response to detection by server 120 of a change in web page content occurring at the web page server 132, 134, or 136. Because the server 120 sends a notification in response to detecting a change in a web page stored at the web server 132, 134, or 136 and not in response to detecting a change in the selection information (or any other information) stored at the server 120, the server 120 is not instructed to generate an alert feed in response to the server 120 detecting a change in state at the server 120.

Furthermore, in the Response to Arguments section of the Final Office Action, the Examiner states that the server 120 "stores a copy of the shortpage and monitors updates to the shortpage according to a user profile." See Final Office Action at page 13. However, this characterization of Tsimelzon is incorrect. Specifically, the server 120 does not store a copy of the shortpage. Rather, the server 120 merely "stores selection information 122 describing shortpages." See Tsimelzon at col. 4, lines 39-41 (emphasis added). Tsimelzon describes that:

¹ Shortpages being pages that include only a subset of elements included on a web page stored on one of the web servers 132, 134, and 136. See Tsimelzon at col. 3, lines 59-66.

² Note that this portion of Tsimelzon describes the block being located using the RepTree. The RepTree is described previously as being part of the selection information stored on server 120. See Tsimelzon at col. 8, lines 9-13.

[w]hen the server receives a request [fro]m the client's browser to create, edit, delete, or view a shortpage or option (element 456), the server translates the request into the appropriate request(s) to the web server(s) (element 458) and retrieves the requested web page(s) or site(s) from the server(s) (element 460). In the described embodiment, these requests are HTTP requests and the web pages are written in HTML or a similar hypertext language. The server does the user-specified action (such as adding selection-enabling information when a shortpage is to be created or edited) or selecting the relevant parts of the web page (when a shortpage is to be viewed) (element 462). See Tsimelzon at col. 6, lines 37-50.

If the server 120 stored a copy of the short page, as alleged by the Examiner, then the server could process a request related to a shortpage without making requests to the web servers. Because the server 120 does not store a copy of the shortpage, the server 120 does not monitor updates to a shortpage stored at the server 120. The change monitored by the server 120 is a change in web page content hosted by the web server 132, 134, or 136. As such, the server 120 is not instructed to generate an alert feed in response to the server 120 detecting a change in state at the server 120 because no such change occurs. Therefore, Tsimelzon does not describe or suggest instructing a remote server to generate an alert feed in response to the remote server detecting a change in state at the remote server.

For at least these reasons, applicants request reconsideration and withdrawal of the rejection of claims 1, 12, and 17, and their dependent claims 3, 6-11, 13-16, 18-20, and 27-38.

Furthermore, the dependent claims recite features that further distinguish Tsimelzon. For example, claims 28, 32, and 36, which depend from claims 1, 12, and 17, respectively, recite that "the state change at the remote server comprises a change to monitored content stored at the remote server." As discussed above, the server 120 of Tsimelzon does not store content. Rather, the server merely stores selection information and retrieves content stored at web page servers 132, 134, and 136 when processing a request. Thus, the server 120 does not experience a change in monitored content stored at server 120.

In addition, as another example, claims 30, 34, and 38, which depend from claims 1, 12, and 17, respectively, recite, *inter alia*, that "instructing the remote server to generate an alert feed comprises instructing the remote server to generate an alert feed upon occurrence of a change in the content." In Tsimelzon, a notification is not generated "upon occurrence" of a change in content. Rather, a notification is generated when a specified time interval has elapsed, the server

120 retrieves the monitored web page, and the server 120 detects a change in content stored at the web page server 132, 134, or 136. Nothing related to notification happens "upon occurrence" of a change in the web page content provided by the web page servers 132, 134, and 136.

For at least these additional reasons, applicants request reconsideration and withdrawal of the rejection of dependent claims 28, 30, 32, 34, 36, and 38.

Claims 21, 22, 24, and 25 each recite "creating a user profile including a request to receive at least one alert corresponding to a state change at the remote server ... instructing the *remote server* to broadcast alert feeds *in response to changes of state at the remote server*" (emphasis added), and claims 23 and 26, as amended, each recite a host configured to "instruct the *remote server* to broadcast alert feeds *in response to changes of state at the remote server*" (emphasis added) and a client configured to "create a user profile including a request to receive at least one alert corresponding to a state change at the remote server." For at least the reasons described above for claims 1, 12, and 17, applicants request reconsideration and withdrawal of the rejection of claims 21-26 because Tsimelzon does not describe or suggest instructing a *remote server* to broadcast alert feeds *in response to changes of state at the remote server*.

Claims 2, 4, and 5 have been rejected as being unpatentable over Tsimelzon in view of Atsmon (U.S. Patent No. 6,607,136). Claims 2, 4, and 5 depend from claim 1. Atsmon does not remedy the failure of Tsimelzon to describe or suggest instructing a *remote server* to generate an alert feed *in response to the remote server detecting a change in state at the remote server*. Accordingly, applicants request reconsideration and withdrawal of the rejection of claims 2, 4, and 5.